Learning Exponential Random Graph Models

Wen Pu
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801
wenpu1@illinois.edu

Jaesik Choi
Computational Research Division
Lawrence Berkeley Laboratory
Berkeley, CA, USA
jaesikchoi@lbl.gov

Eyal Amir
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801
eyal@illinois.edu

Dorothy Espelage
Department of Educational Psychology
University of Illinois at Urbana-Champaign
Urbana, IL 61801
espelage@illinois.edu

Abstract

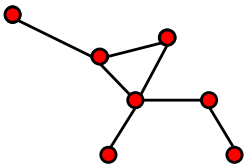
Exponential Random Graphs are common, simple statistical models for social network and other structures. Unfortunately, inference and learning with them is hard for networks larger than 20 nodes because their partition functions are intractable to compute precisely. In this paper, we introduce a novel linear-time deterministic approximation to these partition functions. Our main insight enabling this advance is that subgraph statistics is sufficient to derive a lower bound for partition functions. The proposed method differs from existing methods in the way it exploits asymptotic properties of subgraph statistics. In comparison to current Monte Carlo simulation based methods, the new method is scalable, stable, and precise enough for inference tasks. We show these strengths of the new approach experimentally and theoretically.

1 Introduction

Social network are becoming central to many aspects of life, such as marketing, recruiting, web search, and education programs [16, 5, 18]. Careful use of social network analysis in those areas is key to future advances. For that reason, many researchers and practitioners model their relevant social networks and learn them from data [4]. Many of those social networks are large, and modeling them precisely is hard. Therefore, researchers and practitioners commonly use a family of simple models called Exponential Random Graph Models (ERGM) [22].

An ERGM defines a distribution over all graphs of n nodes. Coefficients and subgraph statistics, such as number of edges, triangles, and k-stars, are then used to specify ERGM distributions [22]. The model captures the correlation of network sub-structures and enables various inferences on complex networks. For example, we can tell whether transitivity is prominent in a network by fitting an ERGM with related subgraphs as features, such as triangles.

Learning ERGMs from data is done by Maximum Likelihood Estimation (MLE). Unfortunately, such learning is hard even for networks of modest size (e.g. 40 nodes) because calculating normalizing constants (*partition functions*) precisely for such models is intractable. For this reason most current techniques involve sampling using Markov Chain Monte Carlo (MCMC) [10, 26]. This results in intractable computation or highly imprecise results for these modest-size-or-larger networks [2, 24, 11, 14].



feature	count	density
edge	7	0.333 (7/21)
triangle	1	0.029 (1/35)
2-star	11	0.105 (11/105)
3-star	5	0.036 (5/140)
rectangle	0	0 (0/70)

Figure 1: An example network of order $n = 7$ (In ERGM, edges are random variables). Table on the right shows the sufficient statistics (densities) for an ERGM with edge, triangle, 2/3-stars and rectangle as features.

In this paper we present a new approximation method for the log partition function of ERGMs. We show that the new method is theoretically and experimentally superior to MCMC in large networks. The new method leads to an overall MLE estimation that is more precise and at the same time scalable. Our advance is primarily in the new partition function approximation; We adapt the MLE to take advantage of the new method.

Specifically, we present a linear-time deterministic approximation to the log partition function of ERGMs. Asymptotic properties of the subgraph statistics space enable this new approximation. The approximation works as follows: Given (coefficient) parameters θ, find that edge-count u (between 0 and $\binom{n}{2}$) that maximizes $\tilde{\gamma}(\theta, u) = \theta^T \rho(u) + C(n, u)$ (See (11) for definition), where $\rho(u)$ is a vector of subgraph statistics approximated for graphs with u edges and function $C(n, u)$ approximates the logarithm of the number of graphs with subgraph statistics close to $\rho(u)$. Once the maximizing u is found, we estimate the log partition function $\ln Z(\theta)$ by $\tilde{\gamma}(\theta, u)$. The approximation works because this $\rho(u)$ captures the subgraph statistics of a large (asymptotically) mass of graphs of n nodes. So, in a sense, many graphs look similar from a subgraph statistics perspective.

We show that the new method performs well experimentally, comparing it to state of the art sampling methods. Our results show that the new algorithm yields reliable approximation when the size of the network is larger than 30.

The rest of the paper is organized as follows. Section 2 reviews ERGM, Section 3 describes the components of the approximation and key theoretical results, Section 4 describes our experimental evaluation, Section 5 describes related work, and Section 6 concludes.

2 Background

An ERGM defines the following distribution over order-n graphs $g \in \mathcal{G}$:

$$p_\theta(g) = \frac{1}{Z(\theta)} \exp\left(\theta^T \phi(g)\right) \tag{1}$$

where $\phi(g)$ is the feature vector for graph $g \in \mathcal{G}$, the parameter θ is a real vector. Partition function $Z(\theta)$ is a normalizing constant, which sums the potentials over $\mathcal{G}$:

$$Z(\theta) = \sum_{g \in \mathcal{G}} \exp\left(\theta^T \phi(g)\right) \tag{2}$$

The feature vector $\phi(g)$ may include any network and nodal attributes of g , and the edge statistics is almost always included [22]. In this work, we focus on undirected graphs and subgraph statistics features for simplicity. Specifically, for a set of subgraph structures of interests $\{L_1, \ldots, L_r\}$, the feature vector of undirected graph g can be defined with subgraph densities as below:

$$\phi(g) = \left(\frac{t(g, L_1)}{t(K_n, L_1)}, \frac{t(g, L_2)}{t(K_n, L_2)}, \ldots, \frac{t(g, L_r)}{t(K_n, L_r)} \right) \tag{3}$$

Here $t(g, L_i)$ counts the number of subgraphs in g that are isomorphic to L_i; K_n is the order-n complete graph, therefore $t(K_n, L_i) = \binom{n}{v_i} t(K_{v_i}, L_i)$ is a constant for any L_i of order v_i.

Example: Figure 1 illustrates a simple example network of order 7. It has seven edges, one triangle, eleven 2-stars, five 3-stars and no rectangle. The third column shows the subgraph densities of the network. For example, the 7-node labeled graph can have at most $\binom{7}{3} \times 1 = 35$ triangles, therefore the triangle density is $1/35 \simeq 0.029$.

Given a network g, the MLE of parameter vector θ is:

$$\theta^* = \operatorname*{argmax}_\theta LL(\theta|g) = \operatorname*{argmax}_\theta \left\{\theta^T \phi(g) - \ln Z(\theta)\right\} \tag{4}$$

In this paper, we are interested in approximating the log partition function $\ln Z(\theta)$.

3 Approximating Log Partition Functions

In this section, we derive a deterministic approximation to the log partition function $\ln Z(\theta)$. We first introduce the counting function for graphs with the same feature vector in Section 3.1, which leads to an efficient approximation in Section 3.2. Section 3.3 reveals a set of edge-number induced lower bounds to $\ln Z(\theta)$ and derives its approximation. Section 3.4 reports the complete algorithm.

3.1 Graph counting in the feature space

We introduce the key concept of graph counting function for the feature space of ERGM. Let $\mathcal{H} = \phi(\mathcal{G})$ be the subgraph density space for $\mathcal{G}$. For $\mathbf{h} \in \mathcal{H}$, we define *counting function* $\#(\mathbf{h}) = |\{g \in \mathcal{G} | \phi(g) = \mathbf{h}\}|$, i.e. the number of graphs in $\mathcal{G}$ having $\mathbf{h}$ as subgraph densities. We re-write the partition function (2) into a compact form using counting function:

$$Z(\theta) = \sum_{\mathbf{h}\in\mathcal{H}} \#(\mathbf{h}) \exp\left(\theta^T \mathbf{h}\right) = \sum_{\mathbf{h}\in\mathcal{H}} \exp\left(\theta^T \mathbf{h} + \ln \#(\mathbf{h})\right) \tag{5}$$

Notice that when $\theta = 0$, each term in (5) simply counts the graphs with given subgraph configuration, and the normalizing constant becomes the total number of graphs $|\mathcal{G}|$. Later we will show how the graph counting interpretation helps in computing $\ln Z(\theta)$.

Let $L_1, L_2, \ldots, L_r$ be simple graphs of interests and v_i be the number of nodes for L_i. The following lemma provides an upper bound to $|\mathcal{H}|$. Under the assumption $\forall i,\ n \gg v_i$ and $n \gg r$, the lemma establishes reasonable error bounds for several arguments in the rest of the paper:

Lemma 1. *For $v* = \max\{v_1, \ldots, v_r\}$, it holds that $\ln |\mathcal{H}| \le r v^* \ln n$.*

Proof. Subgraph count for L_i in any g is bounded by $0 \le t(g, L_i) \le t(K_n, L_i) \le \binom{n}{v_i} v_i!$, therefore

$$\ln |\mathcal{H}| \le \ln \prod_{i=1}^{r} t(K_n, L_i) \le \ln \left[\prod_{i=1}^{r} \binom{n}{v_i} v_i!\right] \le r \ln \left[\binom{n}{v^*} v^*!\right] = r \ln \frac{n!}{(n-v^*)!} \le r v^* \ln n$$

□

3.2 Approximation of Log-Sum-of-Exponentials

Given some set $\mathbf{S}$, and any function $f : \mathbf{S} \to \mathbf{R}$, formula of the form $\ln \sum_{x\in S} \exp f(x)$ can be approximated by $\max_{x\in\{\mathbf{S}\}} f(x)$ if $|S|$ is small. Specifically, we have the following upper and lower bounds:

Lemma 2. *Let f be a function on S and $x^* = \operatorname{argmax}_{x\in S} f(x)$, it holds that:*

$$f(x^*) \le \ln \sum_{x\in S} \exp f(x) \le f(x^*) + \ln |S|$$

Proof. On the lower bound: $f(x^*) = \ln \exp f(x^*) \le \ln \sum_{x\in S} \exp f(x)$. On the upper bound: $\ln \sum_{x\in S} \exp f(x) \le \ln |S| \exp f(x^*) = f(x^*) + \ln |S|$. □

Direct application of Lemma 2 to $\ln Z(\theta)$ yields a sloppy approximation because the huge size of $\mathcal{G}$. Thanks to Lemma 1, the following approximation to (5) has a much tighter error bound:

$$\ln Z(\theta) = \max_{\mathbf{h}\in\mathcal{H}} \{\theta^T \mathbf{h} + \ln \#(\mathbf{h})\} + O(\ln n) \tag{6}$$

In next section, we discuss how to estimate the first term of (6).

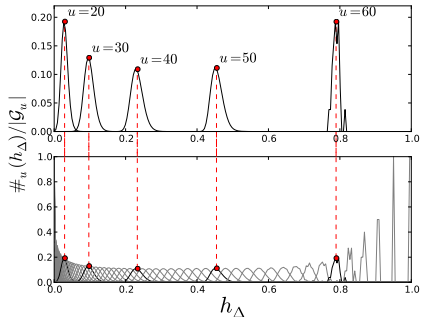


Figure 2: Concentration of triangle density h_Δ conditioned on the number of edges u for unlabeled graphs ($n = 12$). In this case, there are $\binom{12}{2} = 67$ possible edge counts. Y-axis measures the counting function $\#_u(h_\Delta)$ normalized by $|\mathcal{G}_u|$. Lower plot illustrates all 67 distributions; upper plot shows a subset for $u \in \{20, 30, 40, 50, 60\}$.

3.3 Edge-Count Induced Lower Bounds

In this section, we derive an alternative representation to the approximation in (6). Let $\mathcal{G}_u \subset \mathcal{G}$ be the set of graphs with u edges, $\mathcal{H}_u \subset \mathcal{H}$ be the set of subgraph statistics induced by $\mathcal{G}_u$, and $\#_u(\mathbf{h})$ be the restricted counting function which only counts graphs in $\mathcal{G}_u$, i,e. $\#_u(\mathbf{h}) = |\{g \in \mathcal{G}_u | \phi(g) = \mathbf{h}\}|$. For any θ and u, we have the following lower bound to (6):

$$\gamma(\theta, u) = \max_{\mathbf{h} \in \mathcal{H}_u} \{\theta^T \mathbf{h} + \ln \#_u(\mathbf{h})\} \leq \max_{\mathbf{h} \in \mathcal{H}} \{\theta^T \mathbf{h} + \ln \#(\mathbf{h})\} \tag{7}$$

Notice that the equality holds when K_2 (i.e. two nodes with a single edge) is one of the feature subgraphs, because in this case h_{K_2} is uniquely specified by u. Therefore, $\mathcal{H}_u \cap \mathcal{H}_{u'} = \emptyset$ if $u' \neq u$, and $\#_u(\mathbf{h}) = \#(\mathbf{h})$. Specifically:

$$\max_u \{\gamma(\theta, u)\} = \max_u \left\{ \max_{\mathbf{h} \in \mathcal{H}_u} \{\theta^T \mathbf{h} + \ln \#_u(\mathbf{h})\} \right\} = \max_{\mathbf{h} \in \mathcal{H}} \{\theta^T \mathbf{h} + \ln \#(\mathbf{h})\} \tag{8}$$

In practice, K_2 is almost always included as a feature in ERGMs [22]. Therefore, $\max_u \{\gamma(\theta, u)\}$ can be treated as an alternative representation of the approximation in (6), which acts as a tight lower bound to $\ln Z(\theta)$. For the rest of the section, we show that $\gamma(\theta, u)$ can be approximated by exploiting the asymptotic property of $\#_u(\mathbf{h})$ in $\mathcal{G}_u$.

3.3.1 Concentration of subgraph statistics in $\mathcal{G}_u$

In this section, we explain the main intuition that leads to the approximation of $\gamma(\theta, u)$.

Gilbert-Erdös-Rényi random graphs [9, 7] are the widely used probabilistic models for graphs. There are two closely related definitions, $G(n, p)$ by [9] and $G(n, M)$ by [7]. In $G(n, p)$, an order-n graph is generated by drawing each edge independently with probability p; In $G(n, M)$, an order-n graph is chosen uniformly at random from $\mathcal{G}_M$, i.e. the set of all graphs with n nodes and M edges.

Nowicki [21] proved that that $\phi(g)$ is asymptotically normally distributed for $g \in G(n, p)$. Using Chebyshev's inequality, the following lemma extends that result to characterize $\#_u(\mathbf{h})$ in $G(n, M)$ over $\mathcal{G}_u$:

Lemma 3. *Let s_i be the edge count of L_i, define function $\rho_i(x) = (x/\binom{n}{2})^{s_i}$. Given any edge density μ, write the edge count $u = \binom{n}{2}\mu$ as a function of n. Then for any real vector $\mathbf{a} = (a_1, a_2, \ldots, a_r)^T$ and random graph $g \in G(n, M = u)$, the following holds as $n \to \infty$:*

$$P\left(\left|\mathbf{a}^T(\phi(g) - \rho(u))\right| \geq \frac{1}{cn}\right) \to 0 \tag{9}$$

where $\rho(u) = (\rho_1(u), \ldots, \rho_r(u))^T$ and c is some constant.

The proof is available in the appendix. Notice here $\rho_i(u)$ is the expected density of L_i in $G(n, p = u/\binom{n}{2})$. Lemma 3 suggests that any linear combination of $\mathbf{h} = \phi(g)$ tends to concentrate around $\rho(u)$. In a sense, graphs in $\mathcal{G}_u$ forms a cluster in terms of the subgraph statistics. Figure 2 illustrates the phenomenon using order 12 unlabeled graphs [3].

3.3.2 Estimating Lower Bounds

In this section, we derive and analyze an estimation to the lower bound $\gamma(\theta, u)$.

Let $\mathbf{h}'$ and $\mathbf{h}^*$ be the optimum of $\gamma(\theta, u)$ and maximizer of $\#_u(\mathbf{h})$ respectively:

$$\mathbf{h}' = \operatorname*{argmax}_{\mathbf{h}\in\mathcal{H}_u}\{\theta^T\mathbf{h} + \ln \#_u(\mathbf{h})\} \quad \text{and} \quad \mathbf{h}^* = \operatorname*{argmax}_{\mathbf{h}\in\mathcal{H}_u}\{\ln \#_u(\mathbf{h})\}$$

Then the following bounds of $\gamma(\theta, u)$ hold for all θ and u:

$$\theta^T\mathbf{h}^* + \ln \#_u(\mathbf{h}^*) \le \gamma(\theta, u) = \theta^T\mathbf{h}' + \ln \#_u(\mathbf{h}') \le \theta^T\mathbf{h}' + \ln \#_u(\mathbf{h}^*) \tag{10}$$

Notice the gap between upper and lower bounds in (10) is $\theta^T(\mathbf{h}' - \mathbf{h}^*)$. Naturally $\theta^T\mathbf{h}^* + \ln \#_u(\mathbf{h}^*)$ can serve as an approximation to $\gamma(\theta, u)$ within the error of the gap. Three questions still remain: How to compute $\ln \#_u(\mathbf{h}^*)$; How to identify $\mathbf{h}^*$; And how good is the approximation?

For the first question, the following lemma proposes an approximation to the graph counting at $\mathbf{h}^*$:

Lemma 4. *Given edge count u, it holds that*

$$\ln \#_u(\mathbf{h}^*) = \binom{n}{2} H(u/\binom{n}{2}) - O(\ln n)$$

where $H(x) = -x \ln x - (1 - x)\ln(1 - x)$.

The proof is available in the appendix. The intuition of Lemma 4 is to approximate the graph counting at $\mathbf{h}^*$ with $|\mathcal{G}_u|$ using Stirling's approximation, while using Lemma 1 to bound the error in $O(\ln n)$.

For the second one, because $\mathbf{h}^*$ is the maximizer of $\#_u(\mathbf{h})$ in $\mathcal{H}_u$, Lemma 3 suggests $\rho(u)$ as an approximation when n is large. Together with Lemma 4, we propose the following approximation of $\gamma(\theta, u)$:

$$\tilde{\gamma}(\theta, u) = \theta^T\rho(u) + \binom{n}{2} H(u/\binom{n}{2}) \tag{11}$$

To discuss the behavior of the approximation as $n \to \infty$, we again represent edge count $u = \binom{n}{2}\mu$ as a function of n and edge density μ.

Theorem 1. *Assume K_2 is included as a subgraph feature, treat the edge count $u = \binom{n}{2}\mu$ as a function of μ and n, the following holds as $n \to \infty$:*

$$\left|1 - \frac{\ln Z(\theta)}{\max_\mu \tilde{\gamma}(\theta, u)}\right| \to 0$$

Proof. Notice that $\tilde{\gamma}(\theta, u)$ is in $O(n^2)$. For any θ and $x, y \in \mathcal{H}$, we have $|\theta^T(x - y)| \le \sum_{i=1}^r |\theta_i|$. Moreover, when K_2 is included as a subgraph feature, we have shown that $|\ln Z(\theta) - \max_u \gamma(\theta, u)| \le O(\ln n)$ (See (6) and (8)). Together with Lemma 4, we have:

$$\lim_{n\to\infty}\left|\frac{\max_\mu \tilde{\gamma}(\theta, u) - \ln Z(\theta)}{\max_\mu \tilde{\gamma}(\theta, u)}\right| \le \frac{\sum_{i=1}^r |\theta_i| + O(\ln n)}{O(n^2)} \to 0$$

□

3.4 Approximate Algorithm

The estimation of edge-count induced lower bound immediately leads to an approximation of $\ln Z(n)$: **Edge Count Search (ECS) approximation:**

$$\mathrm{ECS}(\theta) = \max_{0\le u\le\binom{n}{2}} \left\{\theta^T\rho(u) + \binom{n}{2} H(u/\binom{n}{2})\right\} \tag{12}$$

Algorithm 1 reports a straightforward implementation of (12), which simply searches through all the u to maximize $\tilde{\gamma}(\theta, u)$. Notice that the algorithm requires no extra parameters, which makes the ECS approximation very easily to use compared to current MCMC sampling methods.

Assume the number of subgraph features $r \ll n$, the time complexity of Algorithm 1 is in $O(n^2)$, which is linear in terms of the number of random variables (i.e. edges) of the model.

Algorithm 1 Our new ECS Approximation to the log partition function $\ln Z(\theta)$

Input: model parameter θ and number of nodes n
Output: estimation of $\ln Z(\theta)$
Initialize $ECS \leftarrow -\infty$
for $u \leftarrow 0$ **to** $n(n-1)/2$ **do**
  $ECS \leftarrow \max\{\tilde{\gamma}(\theta, u), ECS\}$
end for

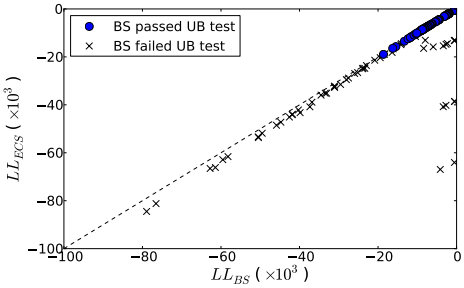


Figure 3: Scatter plot of log-likelihood estimations for ECS and BS on networks of $n = 160$. Many BS estimations fail UB test (13). Otherwise, ECS and BS estimations are very close (top right).

4 Experimental Results

In this section, we use two tasks to evaluate the performance of ECS approximation: estimating log-likelihood functions and MLE estimation. We implement the commonly used triad model (edge, 2-star, triangle) for the experiments.

The intractability of ERGM log partition function for large n makes it impossible to find ground truth, because the asymptotic property used by ECS applies to large n. Instead, we resort to comparing the output of ECS with the state of the art MCMC sampling algorithm for ERGMs: Bridge Sampling [8, 10, 14]. We use the Bridge Sampling implementation (BS) of the widely used R package *statnet* [10] to perform the evaluation. For each trial, we set the number of bridge distributions to 20, the burn-in to 50,000, and the sample size to 20,000. Because *statnet* use raw subgraph counts instead of densities as feature vectors, we properly scaled the parameters to maintain the consistency. *statnet* also provides routines for sampling graphs from a given ERGM, which we used to generate synthetic data set. Notice that our target of sampling is not perfect observations from the given θs, but to diversify the synthetic networks data set.

To alleviate the interference of the well known stability problem from sampling based methods on ERGMs [11, 2], we employ the following upper bound to the log likelihood function as an indicator of bad approximation:

$$LL(g|\theta) = \theta^T \phi(g) - \ln Z(\theta) \leq \theta^T \phi(g) - \max\{0, \sum_{i=1}^{r} \theta_i\} \tag{13}$$

The bound holds for any θ and g, because $\ln Z(\theta)$ must be larger than the log potential of empty graph, which is 0, and of complete graph, which is $\sum_{i=1}^{r} \theta_i$. Notice that by design, ECS will never violate the bound. Because for any θ, we have $\gamma(\theta, 0) = \tilde{\gamma}(\theta, 0)$ and $\gamma(\theta, \binom{n}{2}) = \tilde{\gamma}(\theta, \binom{n}{2})$. We apply this upper bound test (UB test) for all log-likelihood estimations.

4.1 Estimating log-likelihood functions

We sample synthetic networks from a wide range of parameters to evaluate their log-likelihoods. We first generate a $6 \times 6 \times 6$ grid of θ ranging from (-5.0, -5.0, -5.0) to (5.0, 5.0, 5.0), and drop the tuples in which all values have the same sign. We ended up with 162 different θs. Then for each θ, we sampled networks for different $n \in \{30, 40, 60, 80, 100, 120, 140, 160\}$. The total number of sampled graphs is 1,296, and we estimate the log-likelihood for each sampled network using both Bridge Sampling and ECS.

Figure 3 reports the scatter plot of the results of both methods for $n = 160$. Points close to the dashed line suggest ECS and BS produce similar results; Points far away from the dashed line suggests the estimation results are very different. For each estimation of BS, we also check whether it exceeds the UB test. If the estimation exceeds the log-likelihood upper bound, we mark the data point with a cross ($\times$); Otherwise we mark with a blue circle.

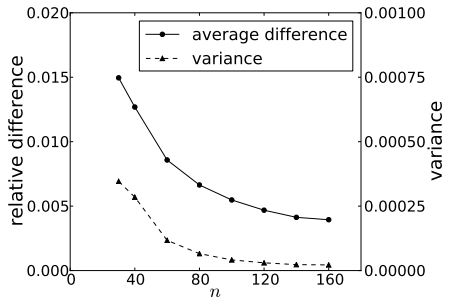


Figure 4: Relative difference between the estimations of ECS and BS for different n, given BS estimation passes the UB test (13). x-axis is the order of the network, y-axis measures the mean and variance of relative differences.

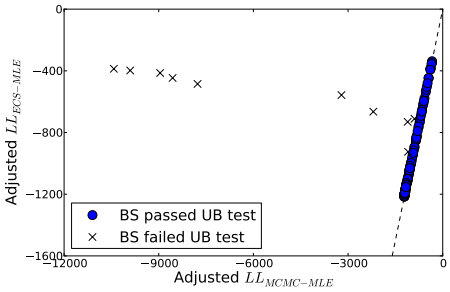


Figure 5: Scatter plot of adjusted log-likelihood for MCMC-MLE and ECS-MLE estimations on networks of $n = 60$. ECS-MLE outperforms MCMC-MLE in all trials (all points are on the left of the dashed line). BS failed UB test (13) in many trials that MCMC-MLE and MCS-MLE significantly disagree.

From 3 we can tell when BS estimation fails the UB test, the difference between ECS and BS results are almost negligible. However, there is a significant portion (about 30%) of BS estimation results turn out to be unrealistic, while ECS keep producing results consistent to (13).

To further compare ECS estimations with the legit BS estimations, we report their relative differences for models that BS estimation pass the upper bound test: $\text{reldiff} = |(LL_{ECS} - LL_{Bridge})/LL_{Bridge}|$. Figure 4 reports the mean and variance of the relative difference for networks of which BS estimation passes the UB test. The plot shows both the mean and variance decrease as n increases. As n increases, the estimations become very close.

4.2 MLE estimation

In this section, we use ECS as a sub-routine to perform MLE estimations on network data. Because the number of subgraph features in triad model is $r = 3$, it is practical to perform grid search over a restricted sub-space of θ. Using both synthetic data and real social networks, we compare the performance of this simple ECS-MLE with MCMC-MLE [24], which uses Bridge Sampling as a sub-routine [10].

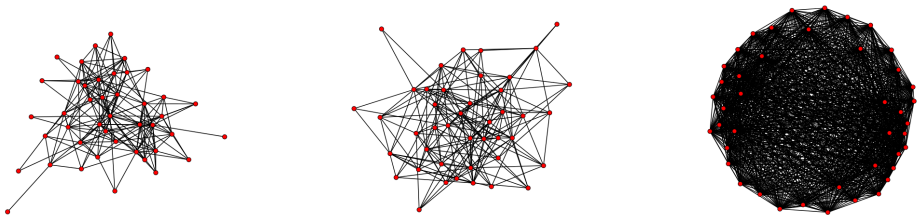
We first generated a $6 \times 6 \times 6$ grid of θ ranging from $(-3.0, -3.0, -3.0)$ to $(3.0, 3.0, 3.0)$. For each θ, we sampled one network of $n = 60$. Then we fit the triad model with the sampled network using both MCMC-MLE and ECS-MLE. For ECS-MLE, we performed grid search in a slightly enlarged parameter space with finer granularity.

To evaluate, we estimate the log-likelihood of the network on both fitted models using Bridge Sampling. As we observed in Figure 3, BS tends to generate unrealistic estimations. If a BS estimation fails the UB test, we take the upper bound as the log-likelihood estimation instead. Notice that the adjusted value is still valid for the purpose of comparing two MLE algorithms. Figure 5 reports the scatter plot for the adjust log-likelihood for both ECS-MLE and MCMC-MLE, showing that ECS-MLE outperforms or is on par with MCMC-MLE in all trials.

We fit the triad model with Kapferer2 network [10] (Figure 6a) to showcase the stability of ECS-MLE. Figure 6b and 6c show the simulated networks from the models learned with ECS-MLE and MCMC-MLE respectively. We set the starting state of the simulation as a randomly sampled network with edge density 0.5 and burn-in to 200,000. The near complete graph simulation of MCMC-MLE suggests the sampling algorithm may be trapped into some local optimum, disregards our extensive efforts on parameter tuning.

5 Related Work

Modeling social network structures has been actively studied in machine learning community. Latent variable models, such as matrix factorization [13], block modeling [1, 15, 12] and others [19, 17],



(a) Original network **(b)** ECS-MLE result simulation **(c)** MCMC-MLE result simulation

Figure 6: Experiments on Kapferer2 data set. 6a is the original network. Network 6b simulated from model learned with ECS approximation outperforms the network 6c simulated from model learned with MCMC-MLE.

represent the relational data with latent variables. Among those, Ho et al. [12] proposed triangular motifs as network representation, which is closely related to ERGM's subgraph features. In comparison, ERGM posts a simple model with intuitive feature specifications that fits for many network analysis tasks.

Computing normalizing constants for complex and high-dimensional models, such as ERGMs, is intractable. Markov chain Monte Carlo simulations are arguably among the most effective methods. Gelman and Meng [8] proposed the path sampling formulation to unify acceptance ratio method and thermodynamic integration from theoretical physics for estimating the (ratios of) normalizing constants. Annealed importance sampling (AIS) [20], which is popular in deep learning literature [23], can also be viewed as one form of thermodynamic integration. Although effective in many applications, Bhamidi et al. [2] shows that the mixing time for any local Markov chain in low temperature regimes of ERGMs is exponentially slow, rendering these methods computationally intractable in many cases. In comparison, ECS approximation is deterministic, therefore avoids the sampling completely.

ECS approximation is a variational inference algorithm. In this category, there are many other techniques, such as pseudo-log-likelihood [25], mean field approximation and Bethe approximation [27]. In the context of ERGM, these methods have been reported to be inferior to sampling based methods [26], and are usually used to generated initial states for sampling based algorithms [14]. ECS distinguishes from others by exploiting the asymptotic property in the feature space of the model. This macroscopic view goes beyond the conditional independence in local structures of the model, and may be more effective for complex high-dimensional models like ERGMs.

ECS approximation is closely related to Chatterjee and Diaconis's work [6]. They apply large derivation principle results on Erdös-Rényi model to derive an analytic approximation to the log-likelihood function of ERGM with non-negative/non-positive parameters for subgraphs (with the exception of K_2). In fact, as $n \to \infty$, (12) converges to their result. Compared to [6], ECS approximation relies on much weaker conditions, therefore more flexible from the algorithmic perspective.

6 Discussion

In this paper, we propose a novel deterministic approximation to the log partition functions of ERGMs. Computing the partition functions (or the ratio of them) is essential in learning ERGMs. Our results show the new method is able to overcome some of the stability issues faced by sampling based methods without losing accuracy. The new algorithm does not depends on extra parameters, making it easy to implement and apply compared to sampling.

We also show that the proposed approximation can be used to build an effective MLE algorithm for ERGMs. In the future, we plan to address various types of MLE problems in EMRGs by using the proposed approximation principles.

References

[1] Edoardo M Airoldi, David M Blei, Stephen E Fienberg, and Eric P Xing. Mixed membership stochastic blockmodels. *The Journal of Machine Learning Research*, 9:1981–2014, 2008.

[2] S. Bhamidi, G. Bresler, and A. Sly. Mixing time of exponential random graphs. In *Proceedings of the 49th IEEE Annual Symposium on FOCS*, 2008.

[3] Andries E. Brouwer. Number of unlabelled graphs with given number of triangles. `http://www.win.tue.nl/~aeb/graphs/cospectral/triangles.html`. Accessed: 2012-09-30.

[4] Peter J Carrington, John Scott, and Stanley Wasserman. *Models and methods in social network analysis*. Cambridge University Press, 2005.

[5] S. Cetintas, M. Rogati, L. Si, and Y. Fang. Identifying similar people in professional social networks with discriminative probabilistic models. In *Proceedings of the 34th ACM SIGIR Conference*, 2011.

[6] S. Chatterjee and P. Diaconis. Estimating and understanding exponential random graph models. *Arxiv preprint arxiv:1102.2650*, 2011.

[7] P. Erdös and A. Rényi. On the evolution of random graphs. *Magyar Tud. Akad. Mat. Kutató Int. Közl*, 5:17–61, 1960.

[8] A. Gelman and X.L. Meng. Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical Science*, pages 163–185, 1998.

[9] E.N. Gilbert. Random graphs. *The Annals of Mathematical Statistics*, pages 1141–1144, 1959.

[10] Mark S. Handcock, David R. Hunter, Carter T. Butts, Steven M. Goodreau, and Martina Morris. *statnet: Software tools for the Statistical Modeling of Network Data*. Seattle, WA, 2003. Version 2.0.

[11] M.S. Handcock. Assessing degeneracy in statistical models of social networks, 2003.

[12] Qirong Ho, Junming Yin, and Eric Xing. On triangular versus edge representations – towards scalable modeling of networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[13] P.D. Hoff. Modeling homophily and stochastic equivalence in symmetric relational data. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.

[14] David R. Hunter, Pavel N. Krivitsky, and Michael Schweinberger. Computational statistical methods for social network models. *Journal of Computational and Graphical Statistics*, 21(4):856–882, 2012.

[15] Charles Kemp, Joshua B Tenenbaum, Thomas L Griffiths, Takeshi Yamada, and Naonori Ueda. Learning systems of concepts with an infinite relational model. In *Proceedings of AAAI Conference*, 2006.

[16] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM SIGKDD Conference*, 2003.

[17] James Robert Lloyd, Peter Orbanz, Zoubin Ghahramani, and Daniel M. Roy. Random function priors for exchangeable arrays with applications to graphs and relational data. In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[18] Juan F Mancilla-Caceres, Wen Pu, Eyal Amir, and Dorothy Espelage. Identifying bullies with a computer game. In *Proceedings of the 26th AAAI Conference*, 2012.

[19] Kurt Miller, Thomas Griffiths, and Michael Jordan. Nonparametric latent feature models for link prediction. *Advances in Neural Information Processing Systems (NIPS)*, 2009.

[20] Radford M Neal. Annealed importance sampling. *Statistics and Computing*, 11(2):125–139, 2001.

[21] K. Nowicki. Asymptotic normality of graph statistics. *Journal of Statistical Planning and Inference*, 21(2):209–222, 1989.

[22] G. Robins, P. Pattison, Y. Kalish, and D. Lusher. An introduction to exponential random graph (p*) models for social networks. *Social Networks*, 29(2):173–191, 2007.

[23] Ruslan Salakhutdinov and Iain Murray. On the quantitative analysis of deep belief networks. In *Proceedings of the 25th ICML*, 2008.

[24] T.A.B. Snijders. Markov chain monte carlo estimation of exponential random graph models. *Journal of Social Structure*, 3(2):1–40, 2002.

[25] D. Strauss and M. Ikeda. Pseudolikelihood estimation for social networks. *Journal of the American Statistical Association*, pages 204–212, 1990.

[26] M.A.J. van Duijn, K.J. Gile, and M.S. Handcock. A framework for the comparison of maximum pseudolikelihood and maximum likelihood estimation of exponential family random graph models. *Social Networks*, 31(1):52–62, 2009.

[27] M.J. Wainwright and M.I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1-2):1–305, 2008.

A Appendix

A.1 Proof of Lemma 3

Before the proof of Lemma 3, we need some preparations. [21] proved that a vector of subgraph counts in $G(n,p)$ are asymptotically normally distributed with a degenerated co-variance matrix with rank 1, as the order of the graph $n \to \infty$. In other words, the subgraph counts are asymptotically linearly dependent on each other. Formally, let $\phi(g') = \{\phi_1(g'), \phi_2(g'), \ldots, \phi_r(g')\}$ be the densities of subgraphs $L_1, L_2, \ldots, L_r$ (i.e. $\phi_i(g') = \frac{t(g', L_i)}{t(K_n, L_i)}$) for $g' \in G(n,p)$, the sizes (number of edges) of these subgraphs are $s_1, s_2, \ldots, s_r$, and $u \sim \text{Bin}(\binom{n}{2}, p)$ is the edge count of g', we have the following theorem:

Theorem 2. *[21] For $g' \in G(n,p)$, and real vector $\mathbf{a} = (a_1, a_2, \ldots, a_r)^T$, the following asymptotic property holds:*

$$n^2 E\left[\mathbf{a}^T\Big(\phi(g') - \rho(u,p)\Big)\right]^2 \to 0 \tag{14}$$

where $\rho(u,p) = (\rho_1(u,p), \ldots, \rho_r(u,p))$, and $\rho_i(u,p) = s_i p^{s_i - 1} \cdot \frac{u}{\binom{n}{2}} - (s_i - 1)p^{s_i}$.

In theorem 2, if we set $p = u/\binom{n}{2}$, then $\rho_i(u, u/\binom{n}{2}) = \left(\frac{u}{\binom{n}{2}}\right)^{s_i}$, which becomes the expected density of L_i in $G(n, p = u/\binom{n}{2})$.

Next step is to extend the above property from $G(n,p)$ to $G(n,M)$.

Corollary 1. *For $g \in G(n, M = u)$, as $n \to \infty$, it holds that*

$$n^2 E_u\left[\mathbf{a}^T\Big(\phi(g) - \rho(u)\Big)\right]^2 \to 0$$

where $\rho_i(u) = \left(u/\binom{n}{2}\right)^{s_i}$

Proof. Following theorem 2, let $\rho_i(u) = \rho_i(u, u/\binom{n}{2})$, as $n \to \infty$, the following holds for $g' \in G(n, p = (u/\binom{n}{2}))$:

$$\begin{aligned}
&n^2 E\left[\mathbf{a}^T\Big(\phi(g') - \rho(u)\Big)\right]^2 \to 0 \\
\Rightarrow &n^2 E\left[E_u\big[\mathbf{a}^T\Big(\phi(g') - \rho(u)\Big)\ |u\big]^2\right] \to 0 \\
\Rightarrow &n^2 \sum_u p(u)\, E_u\big[\mathbf{a}^T\Big(\phi(g') - \rho(u)\Big)\ |u\big]^2 \to 0
\end{aligned}$$

Because $\sum_u p(u) = 1$ and $p(u) > 0$, the claim holds. □

Let c be some positive constant, apply Chebyshev's inequality to the linear combination $\mathbf{a}^T\phi(g)$, we get:

$$P\left(\left|\mathbf{a}^T\left(\phi(g) - E_u\left(\phi(g)\right)\right)\right| \geq \frac{1}{2cn}\right) \leq 4c^2 n^2 \text{Var}(\mathbf{a}^T\phi(g)) \tag{15}$$

Now we start to prove Lemma 3.

Proof of Lemma 3. We first define function $\varepsilon(u)$:

$$\varepsilon(u) = \mathbf{a}^T(E_u(\phi(g)) - \rho(u)) \tag{16}$$

As we know

$$E_u\left(\mathbf{a}^T(\phi(g) - \rho(u))\right)^2 \geq E_u\left(\mathbf{a}^T\left(\phi(g) - E_u(\phi(g))\right)\right)^2 = \text{Var}(\mathbf{a}^T\phi(g)) \tag{17}$$

The equality holds if and only if $\varepsilon(u) = 0$. We can get the following property after applying it to corollary 1: as $n \to \infty$

$$n^2 \text{Var}\left(\mathbf{a}^T\phi(g)\right) \to 0 \tag{18}$$

Therefore, as $n \to \infty$

$$\begin{aligned} & n^2 E_u \left(\mathbf{a}^T \left(\phi(g) - \rho(u) \right) - \varepsilon(u) \right)^2 \to 0 \\ \Rightarrow & n^2 E_u \left(\mathbf{a}^T \left(\phi(g) - \rho(u) \right) \right)^2 - n^2 \varepsilon(u)^2 \to 0 \\ \Rightarrow & |\varepsilon(u)| < \frac{1}{2n} \end{aligned} \tag{19}$$

The last step used corollary 1.

We slacks (15) using (17), and rewrite the inner expectation term using (16):

$$P\left(\left| \mathbf{a}^T \left(\phi(g) - \rho(u) \right) - \varepsilon(u) \right| \geq \frac{1}{2cn} \right) \leq 4c^2 n^2 E_u \left(\mathbf{a}^T \left(\phi(g) - \rho(u) \right) \right)^2 \tag{20}$$

Using (19), we can get

$$P\left(\left| \mathbf{a}^T \left(\phi(g) - \rho(u) \right) - \varepsilon(u) \right| \geq \frac{1}{2cn} \right) \geq P\left(\left| \mathbf{a}^T \left(\phi(g) - \rho(u) \right) \right| \geq \frac{1}{cn} \right)$$

Therefore, apply corollary 1, as $n \to \infty$, we get

$$P\left(\left| \mathbf{a}^T \left(\phi(g) - \rho(u) \right) \right| \geq \frac{1}{cn} \right) \to 0$$

□

A.2 Proof of Lemma 4

Proof of Lemma 4. Let $\mathcal{G}_u$ be the set of graphs with edge count u, since $\mathbf{h}_u^*$ is the maximizer, we have

$$\#(\mathbf{h}_u^*) \geq \frac{|\mathcal{G}_u|}{|\mathcal{H}|}$$

Together with the trivial $\#(\mathbf{h}_u^*) \leq |G_u|$, we can get:

$$\ln |\mathcal{G}_u| - \ln |\mathcal{H}| \leq \ln \#(\mathbf{h}_u^*) \leq \ln |\mathcal{G}_u| \tag{21}$$

Apply Stirling's approximation on $\ln |\mathcal{G}_u|$:

$$\begin{aligned} \ln |\mathcal{G}_u| = & \ln \binom{\binom{n}{2}}{u} \\ \simeq & \left(\binom{n}{2} - u \right) \ln \frac{\binom{n}{2}}{\binom{n}{2} - u} + u \ln \frac{\binom{n}{2}}{u} \\ = & \binom{n}{2} H(u / \binom{n}{2}) \end{aligned} \tag{22}$$

Therefore claims hold. □